CLUBE

Z 80

Novembro /83

N.º 14

COORDENADORES: MARIA IRENE • ALBERTO FERNANDES • AV. BOAVISTA, 832 • TELEF. 65127 • 4100 PORTO

NESTE NÚMERO

ENCICLOPÉDIA DA LINGUAGEM BASIC (Cont.) 1
INTRODUÇÃO À LINGUAGEM MÁQUINA (Cont.) 3
SIMULAÇÃO 6
"DICAS" DE PROGRAMAÇÃO 8

Programas ZX81/Spectrum

Motor de Explosão 9
Hammurabi 12
Gráficos 13
Biorritimo 14
Spectrum Lprint 15
Vu-Meter 16
Conversão 17

MICRODRIVE 18
LIVROS-ACTUALIZAÇÃO DE PREÇOS 19
SOFTWARE 20
PERGUNTAS/RESPOSTAS/SUGESTÕES 21

No Interior:

Cupão de Inscrição

Páginas Centrais:

Esquemas do Circuito Spectrum (Parte II)

Edição: Clube Z80

Fotocomposição: Fotomecânica Mabreu/Porto

Impressão: Gráfica Firmeza/Porto

Tiragem: 500 exemplares, Novembro 1983

ENCICLOPÉDIA DA LINGUAGEM BASIC

NOVEMBRO / 83

Continuamos neste número a tratar do operador lógico AND

Por exemplo, se PRINT A=4 AND B=8; se A é igual a 4 e B igual a 8 o computador deverá imprimir o número −1. Se qualquer das condições falhar, o computador imprimirá o valor 0.

```
PROGRAMA TESTE # 3

10 REM 'TESTE LOGICO....condicao AND'
20 PRINT "NUMERO  de 1  a  10"
30 INPUT A
40 LET B=A >4 AND A <11
50 IF B=0 THEN GOTO 80
60 PRINT A; "  E MAIOR QUE 4 E MENOR QUE 11"
70 GOTO 20
80 PRINT A; " nao e :  MAIOR QUE 4 E MENOR QUE 11 "
100 STOP
```

Amostragem do funcionamento:

RUN
2 (enter ou new line)
Não é maior que 4 e menor que 11
8 (enter ou new line)
Maior que 4 e menor que 11

O OPERADOR AND É USADO POR VÁRIOS MICROCOMPUTADORES PARA EFECTUAREM OPERAÇÕES BINÁRIAS (usando Álgebra de Boole).

DE UMA FORMA LIGEIRA, NÓS PODEMOS DESCREVER ESTE TIPO DE OPERAÇÃO DA FORMA SEGUINTE:

Esta operação compara dois bits (0 ou 1) determinando quando são ambos '1'.

Quando ambos os bits são 'binário 1' o microcomputador responde com o valor '1'.

Por exemplo:

1 and 0 =0
0 and 1 =0
1 and 1 =1

Quando o computador executa a operação AND de um número com outro, executa uma avaliação lógica e produz um terceiro número, que será '0' ou '1'.

Exemplo:

DECIMAL		BINÁRIO
3		0011
	AND lógico	
5		0101
1		0001

Neste exemplo apenas o bit mais à direita é o número 1 em binário; deste modo, o valor resultante será o decimal 1 (binário 0001).

```
PROGRAMA TESTE  #4

10 REM ' TESTE BINARIO  AND '
20 PRINT " VALOR PARA X ? "
30 INPUT X
40 PRINT " VALOR PARA Y ? "
50 INPUT Y
60 LET A = X   AND  Y
70 PRINT " VALOR LOGICO 'AND' para X=";X;" AND ";Y; " e = ";A
80  GOTO 20
100 STOP
```

AMOSTRAGEM DESTE PROGRAMA

RUN
X=6 .... Y=10
VALOR LÓGICO DE 6 and 10=2........ (0010 em binário)

Atenção: Alguns computadores usam ou permitem a fórmula:
A. para AND

No caso do computador WANG 2200 B que está colocado em algumas empresas portuguesas, será necessário estudar a sintaxe do AND porque existem algumas variantes do seu uso, bastante singulares.
Nos casos das máquinas SINCLAIR o AND lógico não funciona como exemplificamos acima, mas sim desta forma:

x AND y = x se y diferente de zero
x AND y = 0 se y é falso (zero)

NOVOS LIVROS

— **20 SIMPLE ELECTRONIC PROJECTS FOR THE ZX81 & SPECTRUM**
ADAMS Stephen, Interface, London, 1982 — **Preço (fotocópias) 300$00**
(Pequenos projectos electrónicos do tipo portas paralelas, facilmente adaptáveis a outras máquinas)

— **INSTANT ARCADE GAMES FOR THE SINCLAIR ZX SPECTRUM**
FROST Jean, PAN, London, 1983 — **Preço (fotocópias) 520$00**
(Um bom auxiliar para aqueles que pretendem construir os s/ próprios jogos. A ideia básica é construir os programas como se se partisse de blocos tipo "Lego". Fornece também conhecimentos de BASIC, nomeadamente para a sua aplicação nos jogos).

— **DYNAMIC GAMES FOR THE ZX SPECTRUM**
HARTNELL Tim, Sinclair Browne, London, 1983 — **Preço (fotocópias) 480$00**
(Cerca de 20 jogos de aventura e acção com explicações introdutórias. Algumas ideias e técnicas que o ajudarão a melhorar e a criar programas seus).

— **MASTER YOUR ZX MICRODRIVE**
PENNEL Andrew, Sunshine Books, London 1983 — **Preço (fotocópias) 350$00**
(Toda a informação necessária para usar os Microdrives. Criação de ficheiros em BASIC. Protecção dos programas).

DESCONTO DE 10% PARA SÓCIOS DO CLUBE

INTRODUÇÃO À LINGUAGEM MÁQUINA

ZX81

Autor: FERNANDO D'ALMEIDA PRECES

(Continuação)

2.ª ROTINA EM CÓDIGO MÁQUINA COM INÍCIO EM 16539

Introduza uma REM:

4 REM (reserva de 21 caracteres)

Em seguida, com a ajuda do monitor já apresentado, meta este código:

```
 33   LD HL, NN
196 }
  2 } número 708 (a 708.ª posição no FICHEIRO)
237 }
 91 } LD DE, NN
 12 }
 64 } 16396 — D. FILE
 25   ADD HL, DE (D. FILE+708)
235   EX DE, HL (coloca a soma em DE)
      NOTA: encontrará explicação para esta troca
            no resumo teórico da instrução LDIR
 33   LD HL, NN }  aponta HL para a primeira
173 }           }  localização dos gráficos
 64 } 16557     }  da nave
  1   CD BC, NN }
  3             }  o contador
  0   (3)       }
237 }
176 } LDIR
201 }
  6 } códigos
128 } da
134 } figura
```

Acrescente:

182 LET C=USR 16539

Apague as linhas 184, 185 e 186
e RUN 150...

Ao observar a imagem, vai notar uma maior nitidez (contraste) no objecto fixo, pois aumentou o tempo de exposição (o número de imagens por segundo em que ele aparece) e a sequencia do conjunto é mais rápida.

Pode no entanto melhorar ainda estas condições, juntando numa só o conteúdo das 2 rotinas, poupando o espaço de tempo da instrução RET no retorno ao BASIC da primeira rotina e a passagem do BASIC pela instrução USR para a 2.ª rotina.

É necessário para tal fazer algumas correcções.
Chame a REM 3 e reserve mais 20 caracteres.

Através do monitor com início em 16532, introduza este código:
33, 196, 2, 237, 91, 12,64, 25, 235, 33, **166, 64,** 1, 3, 0, 237, 176, 201, 6, 128, 134.

No código introduzido a única alteração que se impunha era a do endereço absoluto do primeiro caracter da nave que se encontrava no exemplo anterior em 16557 e neste passou a 16550.

Pode apagar a REM linha 4, e a linha 182.
e RUN 150...

Podemos agora avançar para a rotina que fará executar as 3 linhas que destinamos ao simulcro duma nave. A primeira parte da rotina qua vai de 16514 a 16531 continua a ser igual, pelo que vamos chamar de novo a REM 3 e reservar, para além dos caracteres que esta já tem, mais 35 caracteres.

Através do monitor vamos introduzir o código que se segue:

1.ª localização 16532

```
 33    LD HL, NN
184 }  16568            } localização do 1.º grupo
 64 }                   } de 3 caracteres
229    PUSH HL            coloca no STACK a informação
  1    LD BC, NN
 33 }  (33)               valor para um salto completo de linha
  0 }
 42    LD HL, (NN)
182 }  16566            } localização da posição para a 1.ª linha da nave,
 64 }                   } que será 607+33, após a soma com BC
  9    ADD HL, BC
 34    LD (NN), HL
182 }  16566            } recoloca no mesmo endereço a
 64 }                   } posição para a nova linha
237 }  LD DE, NN
 91 }
 12 }  16396              D. FILE
 64 }
 25    ADD HL, DE
235    EX DE, HL
225    POP HL             vai ao STACK buscar o endereço
  1    LD BC, NN
  3       (3)             (o contador)
  0
237 }  LDIR
176 }
 62    LD A, N          } Este grupo de instruções está atento ao aparecimento
 13    (13) caracter$   } em HL do caracter ($) que significa o fim da imagem a
190    CP, (HL)         } transmitir para saltar para o fim da rotina
 40    JR Z, DIS          Salta 15 instruções se o resultado
 15    (15)               da comparação for 0
229    PUSH HL—coloca o conteúdo de HL no STACK
 24    JR DIS        } Salto de 30 instruções para
226    (256-30)      } trás para recomeçar o ciclo.
 95
  2    (607)
```

0 38 0	1.° grupo de caracteres	
6 128 134	2.° grupo	
0 59 0	3.° grupo	
13	($) final da sequência	
33	LD HL, NN	Quando termina o ciclo, após a transmissão das 3 **fatias** da imagem, volta a repor o número inicial da posição
95 2	607	
34	LD (NN), HL	
182 64	16566	
201	RET	Retorno ao BASIC

A recomendação que inicialmente lhes fiz sobre gravar os programas antes de os correr continua a ser válida, pois por vezes ao utilizarmos a linguagem máquina se houver um pequeno erro num código... lá se vai o nosso trabalho.

De novo RUN 150...

Agora o leitor vai obter uma imagem rápida e bastante mais estável. Apenas o senão dum leve tremido do SCROLL.

Há outros processos para obter ainda uma melhor qualidade na imagem, mas para tal é necessário utilizar outros tipos de instruções mais complexas que ainda teremos oportunidade de abordar em capítulos futuros.

Após a execução desta rotina que exemplifica como colocar uma imagem fixa num movimento SCROLL, em código máquina, o leitor é encorajado a experimentar não só o formar outras imagens no mesmo movimento, mas também a fazê-lo nos restantes movimentos SCROLL.

Para terminar o primeiro capítulo, vamos efectuar a programação de alguns jogos que terão como principal missão a recapitulação de algumas instuções máquina, bem como a introdução de mais umas rotinas da ROM.

(Continua no próximo número)

NOVOS LIVROS

— SPECTRUM MICRODRIVE BOOK

LOGAN Ian, Melbourne, London, 1983 — **Preço (fotocópias) 300$00**

(BASIC. Microdrive. Rede Local. RS232. Código Máquina...).

— EXPLORING SPECTRUM BASIC

LORD Mike, CCP, London — **Preço (fotocópias) 500$00**

(Cerca de 50 programas c/ explicações detalhadas. Pequenas rotinas. Como criar programas. Apêndices c/ informações úteis).

— JEUX ET APPLICATIONS POUR ZX SPECTRUM

HARWOOD David, Interface, London, 1982 — **Preço (fotocópias) 280$00**

(Trad. francesa de «60 GAMES AND APPLICATIONS FOR THE ZX SPECTRUM»).
(Uma colecção de prog. de jogos e utilitários c/ explicações introdutórias).

DESCONTO DE 10% PARA SÓCIOS DO CLUBE

SIMULAÇÃO

PARTE II / NOV. 83

Anteriormente estivemos a estudar o problema que pretendemos resolver através de uma simulação.

Vamos agora decidir sobre as técnicas que usamos em cada um dos aspectos mais importantes deste problema.

Um vez que o problema tem a ver com acções dependentes do tempo, teremos de ver como medimos o tempo (simulado). Para executar esta acção usaremos variáveis HT (tempo/hora) e MT (tempo/minutos). Como unidade e tempo simulado vamos usar 4 minutos, que é o tempo médio para servir um cliente. O nosso programa irá devidir o tempo em segmentos de 4 minutos cada. Durante esse lapso de tempo, o computador leva a efeito algumas tarefas e avança depois para o próximo segmento de tempo, ajustando MT e HT.

Quando MT excede 60 minutos (uma hora), subtraímos 60 e adicionamos 60 à variável TH, que será aumentada em 1. Vamos ver antecipadamente como executamos esta rotina na linha 1000:

```
1000 REM AVANÇO DO TEMPO
1010 LET MT=MT+4
1020 IF TM>=60 THEN GO TO 1030
1025 GO TO 1100
1030 LET TM=TM-60
1040 LET TH=TH+1
1100 RETURN
```

Possuímos dados estatísticos sobre a chegada dos clientes, e esses dados são guardados numa tabela, ou seja, os valores da variável indexada A(H), (H=7,8,9...., 18). O significado de A(7) é o seguinte: o número de clientes que chegam entre as 7 e as 8 horas ficará guardado em A(7). Deste modo, A(18) será o número de clientes que chegam entre as 18 e as 19 horas.

A primeira acção do programa será a de activar a tabela A(H).

```
10 DIM A(18)
20 DATA 30,15,6,3,8,25,9,8,12,12,35,22
30 FOR H=7 TO 18
40 READ A(H)
50 NEXT H
```

Uma observação para aqueles que não possuem DATA e READ na sua máquina — substitua o programa acima descrito por:

```
10 DIM A(18)
20 LET A(13)= 9
21 LET A(14)= 8
22 LET A(15)=12
23 LET A(16)=12
24 LET A(17)=35
25 LET A(18)=22
26 LET A( 7)=30
27 LET A( 8)=15
28 LET A( 9)= 6
29 LET A(10)= 3
30 LET A(11)= 8
31 LET A(12)=25
```

O próximo passo será o de ler os dados referentes aos «clientes impacientes».

Vamos denominar D(K) a percentagem de clientes que partem ao ver o balcão com muitos clientes, isto é, com K clientes.

Denominaremos P(K) o valor estatístico correspondente à percentagem de clientes que entram e ficam. Vamos então ver esta parte do programa.

```
100 DIM D(20)
105 DIM P(20)
110 DATA 0,0,.15,.05,.15,.60,.35,.80,.50.
115 READ D(0),PO)
120 READ D(1),P(1)
130 LET D(2)=D(1):LET D(3)=D(1)
140 LET P(2)=P(1):LET P(3)=P(1)
150 READ D(4),P(4)
160 LET D(5)=D(4):LET D(6)=D(4)
170 LET P(5)=P(4):LET P(6)=P(4)
180 READ D(7),P(7)
190 LET D(8)=D(7):LET D(9)=D(7):LET D(10)=D(7)
200 LET P(8)=P(7):LET P(9)=P(7):LET P(10)=P(7)
210 READ D(11),P(11)
220 FOR J=12 TO 15
230 LET D(J)=D(11):LET P(J)=P(11)
240 NEXT J
```

OBSERVAÇÃO (ZX81 ou TMS 1000)

Para aqueles que possuem uma máquina que não excuta READ e DATA, teremos de alterar o programa no sentido de obter o mesmo efeito.

Deste modo, alteramos as linhas da seguinte forma:

```
110 LET D( 1)=  0
111 LET P( 1)=  0
112 LET D( 2)=.15
113 LET D( 2)=.05
114 LET D( 5)=.25
115 LET P( 5)=.15
116 LET D( 8)=.60
117 LET P(8)=.35
118 LET D(12)=.80
119 LET D(12)=.50
120 LET A=3
121 LET B=4
122 GO SUB 230
123 LET A=6
124 LET B=7
125 GO SUB 230
126 LET A=9
127 LET B=11
128 GO SUB 230
129 LET A=13
130 LET B=16
131 GO SUB 230
132 GO TO 300

230 FOR J=A TO B
232 LET D(J)=D(J-1)
234 LET P(J)=P(J-1)
236 NEXT J
238 RETURN
```

Algumas notas serão necessárias para esclarecer que a alteração do índice das variáveis é imprescindível quando (como com o ZX81 ou TMS 1000) não é aceite o índice zero para variáveis indexadas.

No caso do SPECTRUM, também o índice zero não é aceite, pelo que será útil alterar o índice de D(0) e P(0) para D(1) e P(1) e, consequentemente, incrementar todos os índices de um valor unitário.

O próximo passo do programa é o de colocar o «clock» ou relógio no início do dia.

(Continua no próximo número)

"DICAS" DE PROGRAMAÇÃO

ZX81 / TMS 1000

POUPE MEMÓRIA SABENDO PARA ONDE VAI O PRÓXIMO BYTE...!!

Há várias técnicas para encurtar programas, tendo em vista a economia de memória.
Posto que o ZX81 usa 6 bytes para armazenar numeros, conseguiremos uma economia notável substituindo todos os números de um programa (excepto os de linha) por símbolos equivalentes. Se os números 0 e 1 forem substituídos por NOT PI e SGN PI, respectivamente, obtem-se uma economia de 4 bytes de cada vez.
Para um número maior, de um ou dois algarismos, pode usar-se a função VAL. Por exemplo, VAL "4" poupa 2 bytes, pois o algarismo 4 é armazenado como "string" e não como número.
A função CODE pode ser usada com números maiores, desde que haja um símbolo, no conjunto de caracteres, que corresponda ao número em causa. Por exemplo, CODE "IF" substituirá o número 250.
Se o mesmo número é usado repetidamente ao longo dum programa, então o melhor é atribuí-lo, no início, a uma variável e usá-la de cada vez que for necessário. Se o número 500 aparece repetidas vezes num programa, então poder-se-á atribuí-lo inicialmente a uma variável — digamos, LET N=500. Feito isto, quando surgirem situações do tipo LET X=X+500, passaremos a escrever LET=X+N, poupando 5 bytes de cada vez.
Além da substituição dos valores numéricos, outras técnicas há eventualmente aplicáveis (ou aplicadas).
Um processo de poupar memória é reduzir os gastos com instruções condicionais.
Por exemplo, a instrução IF A=X THEN LET P=P+1 pode assumir a forma LET P=P+(A=X), poupando 6 bytes. Muitas das instruções utilizadas pelos MENUS utilizam a função INKEY como no seguinte caso:

```
10 IF INKEY$="8"THEN LET X=X+1
20 IF INKEY$="5"THEN LET X=X-1
```

Elas podem ser usadas para mover algo, no ecran, para a direita e para a esquerda. Ora, isto pode ser substituído por

```
10 LET X=X+(INKEY$="8")-(INKEY$="5")
```

poupando 20 bytes de cada vez. Se a variável tiver de aumentar ou diminuir mais do que um, então os parentesis deverão ser multiplicados pelo número pretendido, como por exemplo:

```
10 IF INKEY$="6"THEN LET Y=Y+2
20 IF INKEY$="7"THEN LET Y=Y-2
```

pode ser substituído por

```
10 LET Y=Y+VAL"2"*((INKEY$="6")-(INKEY$="7"))
```

poupando 13 bytes.

De igual modo, muitas das instruções PRINT podem ser modificadas para economizar memória. Por exemplo, ao escrever as instruções para operar com um programa, certas palavras podem ser substituídas por uma instrução de 1 byte. Será o caso de PRINT "ENTRADA DO SEU NOME", que pode transformar-se no equivalente PRINT "INPUT O SEU NOME" em que "input" é metido como uma instrução. Isto poupa 6 bytes, uma vez que os espaços antes e depois de "input" não custam nada.
Muitas outras palavras-instrução, tais como IF, OR, TO, AND, THEN e outras podem ser usadas como este fim (e são-no). Contudo, para permitir que algumas delas entrem pode ser necessário meter primeiro THEN, e apagá-lo posteriormente.
Quando se vai escrever um texto de linhas que se completam, tal como

```
10 PRINT".......''
20 PRINT".......''
30 PRINT".......''
```

este pode ser substituído por

```
10 PRINT".....",".........","........"
```

o que, neste caso, poupa 6 bytes.

É de lembrar ainda que GOTO e GOSUB não necessitam de ser atribuídos a valores numéricos como GOTO 100, mas que podem usar funções. Por exemplo, a rotina

```
10 IF X=1 THEN GOTO 100
20 IF X=2 THEN GOTO 200
30 IF X=3 THEN GOTO 300
```

pode ser substituída simplesmente por

```
10 GOTO 100*X
```

poupando a enormidade de 40 bytes.

Recorrendo a técnicas como as apontandas, encurtar-se-ão os programas, com economia de memória, e tornar-se-á a programação mais eficaz (por outro lado, as mesmas podem facilitar a análise de determinados passos de programas que a elas recorrem).

RAMIRO VERÍSSIMO

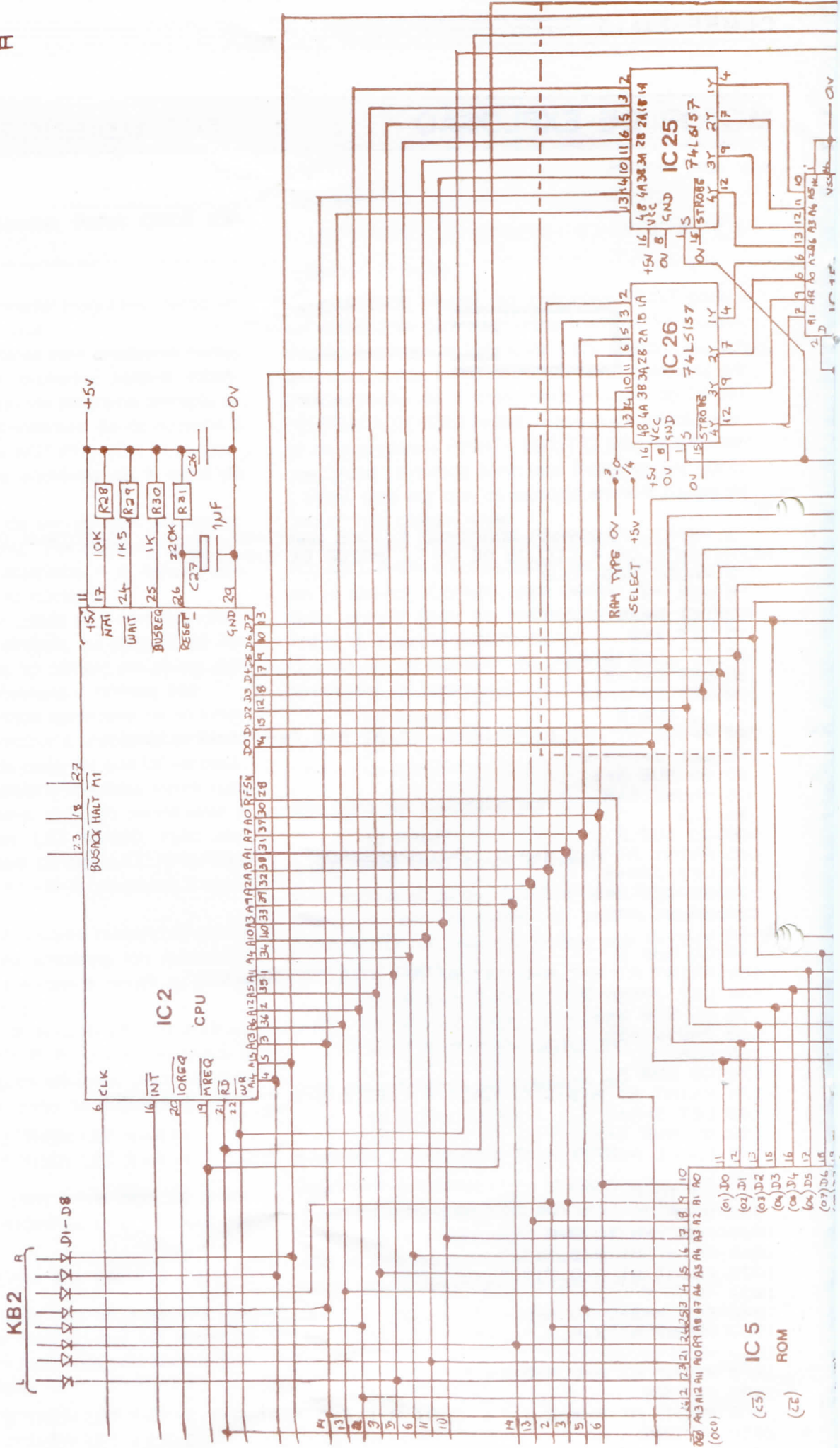
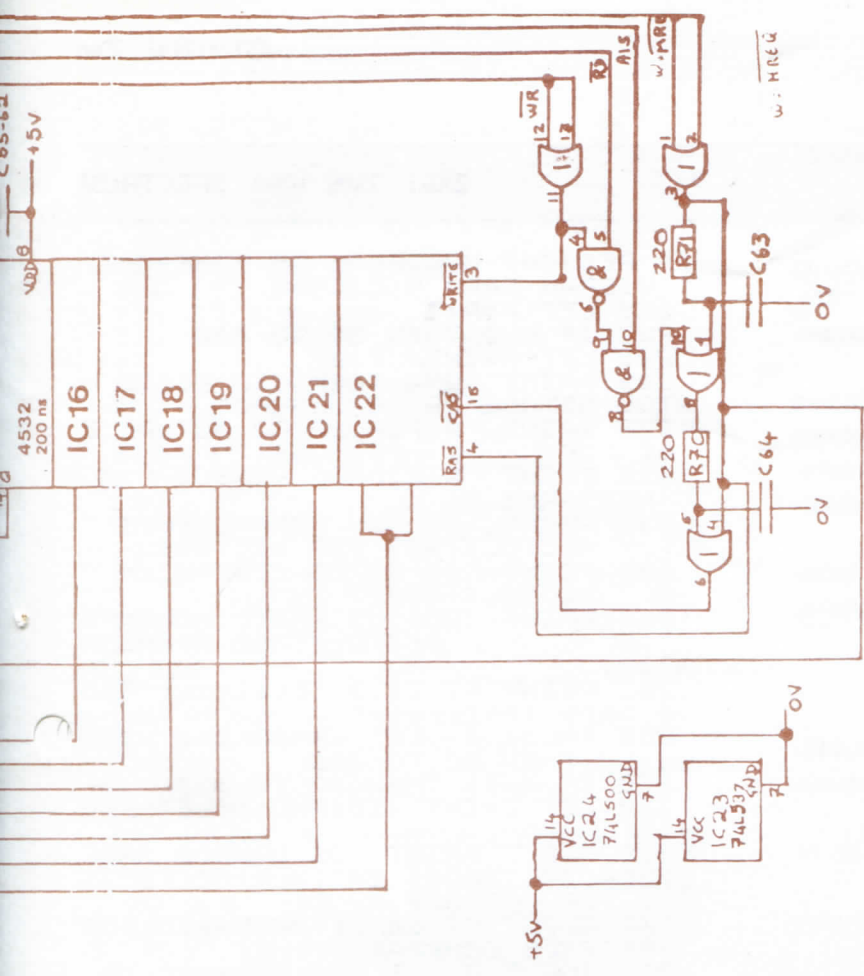
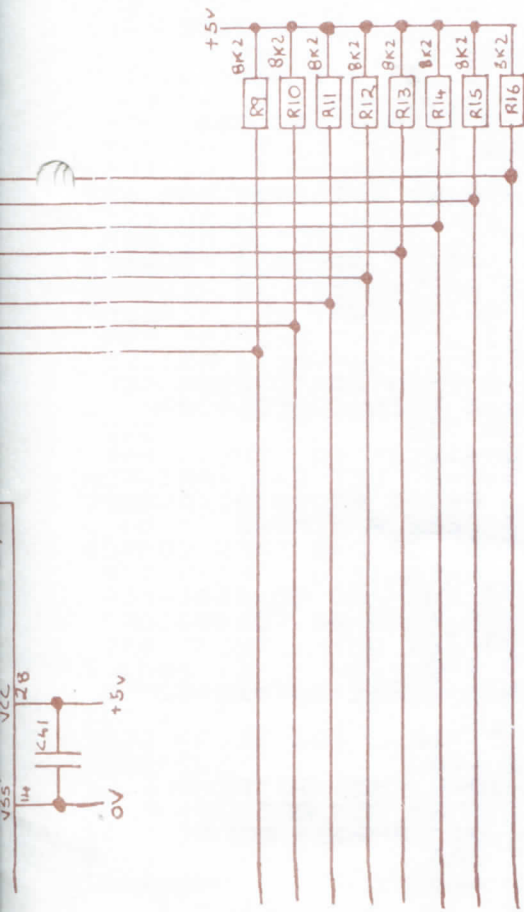
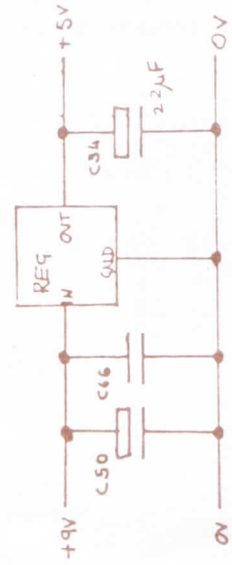
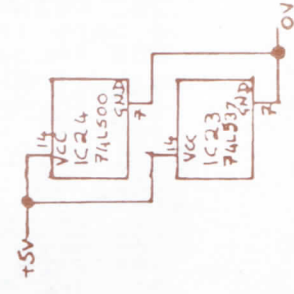
MOTOR DE EXPLOSÃO

SPECTRUM / ZX81

Autor: JOSÉ MARTINS

AR+GAS>- =ADMISSAO

```
   1 PRINT "PROGRAMA ADAPTADO P/JOSE MARTINS";AT 4,0;"SPECTRUM OU ZX81";AT 6,0;"
REPRESENTA OS 4 CICLOS DE UM    MOTOR DE EXPLOSAO"
   2 PAUSE 400: CLS
   5 REM "X"
  10 LET A=10
  15 LET B=2
  20 LET C=B+B/B
  25 LET D=C-B
  30 LET S=A*A*A
  50 GO SUB S
  60 PRINT AT A-B,D;"AR+GAS>- ";AT A-B,A+A;"1=ADMISSAO"
  70 LET J=A+A-B
  80 GO SUB S+S
  85 PAUSE A*A
  90 CLS
 100 GO SUB S
 105 PRINT AT A-C,A+A-D;"2=COMPRESSAO"
 110 LET J=A+B
 120 GO SUB S+S
 125 PAUSE A*A
 140 CLS
 145 GO SUB S
 150 PRINT AT A-C,A+D;"*";AT A-C,A+A;"3=EXPLOSAO"
 154 LET J=A+A-B
 155 GO SUB S+S
 160 PAUSE A*A
 165 CLS
 170 GO SUB S
 175 PRINT AT A-B-B,C;"GAZ< - ";AT A-B-B,A+A-B;"4=ESCAPE"
 180 LET J=A+C
 190 GO SUB S+S
 195 PAUSE A*A
 200 CLS
 210 GO TO A
 999 STOP
1000 FOR Y=A TO A+A
1010 PRINT AT A/B,Y;" "
1020 PRINT AT A-D,Y;" "
1030 NEXT Y
1040 FOR X=B+C TO A-D
1050 PRINT AT X,A;" "
1060 NEXT X
1070 PRINT AT A-C,A-D;"- "
1100 RETURN
2000 PRINT AT A-B-B,J;"    ";AT A-B,J;" ";AT A-B,J;"    "
2050 RETURN
```

4532
200 ns
VDD
+5V
IC16
IC17
IC18
IC19
IC20
IC21
IC22
RAS
CAS
WRITE
WR
RD
A15
W.MREQ
220
R70
R71
C63
C64
0V
VCC
IC24
74LS00
GND
IC23
74LS37

OPTIONAL 32K RAM

HAMMURABI

ZX81 / TMS 1000 / SPECTRUM

Trata-se de um jogo de simulação social. Está escrito em BASIC e conhecemos este programa desde 1974. Foi originalmente escrito em FOCAL para um computador DEC por autor desconhecido.

A ideia de o publicar (não contém gráficos) destina-se mais a mostrar que todos os programas de aventuras giram à volta da simulação social e, adicionalmente, podem existir imagens que são chamadas em determinadas situações (e que o leitor pode criar).

Se os leitores pretenderem modificar algum dos factores, o uso intensivo de REMs, ao longo do programa, poderá ajudar a execução dessa tarefa.

DESCRITIVO DA ACÇÃO

O jogador é HAMMURABI, o administrador da SUMÉRIA. A cidade, inicialmente, possui 1000 hectares, 100 pessoas e 3000 hl. de trigo nos celeiros.

Para comprar terras, o preço varia entre 17 e 26 hl. de trigo por hectare.

Usará esse trigo para semear no próximo ano e para alimentar a população da cidade.

Rapidamente se verifica que um certo número de pessoas apenas pode ter uma certa área de terra e que as pessoas desaparecem se não forem alimentadas convenientemente. Podem também aparecer situações de crise, originadas por pragas ou ratos que destróem o cereal armazenado.

Resumindo e concluindo, gerir os poucos recursos não é tarefa fácil, digamos, por um período de 10 anos.

```
ESTE PROGRAMA PODE SER USADO NO
ZX81 OU TMS/1000,desde  que use
apenas uma instrucao por linha;
  por exemplo  :

linha 100 let z=0:let p=95:let s
=28:let h=3000:let e=h-s

sera transformada em

100 LET Z=0
102 LET P=95
104 LET S=2800
106 LET H=3000
108 LET E=H-S
   5 CLEAR
  10 PRINT TAB 32;"HAMURABI"
  30 PRINT : PRINT : PRINT
  80 PRINT "TENTE GOVERNAR A ANT
IGA SUMERIA"
  90 PRINT "NUM PERIODO DE 10 AN
OS.": PRINT
  95 LET D1=0: LET P1=0
 100 LET Z=0: LET P=95: LET S=28
00: LET H=3000: LET E=H-S
 110 LET Y=3: LET A=H/Y: LET I=5
: LET Q=1
 210 LET D=0
 215 PRINT : PRINT "HAMURABI: DE
VO INFORMA-LO,": LET Z=Z+1
 217 PRINT "QUE NO ANO ";Z;",";D
;" PESSOAS MORRERAM DE FOME,";I;
" VIERAM PARA A CIDADE."
 218 LET P=P+I
 227 IF Q>0 THEN GO TO 230
 228 LET P=INT (P/2)
 229 PRINT "UMA TERRIVEL PESTE M
ATOU METADE DA POPULACAO"
 230 PRINT "A POPULACAO ACTUAL E
DE ";P;"         PESSOAS."
 232 PRINT "A CIDADE POSSUI ";A;
" HECTARES."
 235 PRINT "COLHEM NORMALMENTE "
;Y;" HECTOLITROS POR HECTARE."
 250 PRINT "OS RATOS DESTROIEM "
;E;" HECTOLITROS."
 260 PRINT "POSSUI NESTE MOMENTO
 ";S;"          HECTOLITROS ARMAZEN
ADOS."
 261 PRINT AT 21,0;"Qualquer Tec
la para continuar"
 262 PAUSE 0: LET y$=INKEY$: IF
y$="" THEN GO TO 262
 270 IF Z=11 THEN GO TO 860
 310 LET C=INT (10*RND): LET Y=C
+17
 312 CLS : PRINT "OS CAMPOS SAO
COMERCIALIZADOS EM";Y;" HECTOLIT
ROS POR HECTARE."
 320 PRINT AT 18,0;"HECTARES QUE
 PRETENDE COMPRAR ?";
 321 INPUT Q: IF Q<0 THEN GO TO
850
 322 IF Y*Q<=S THEN GO TO 330
 323 GO SUB 710
 324 GO TO 320
 330 IF Q=0 THEN GO TO 340
 331 LET A=A+Q: LET S=S-Y*Q: LET
 C=0
 334 GO TO 400
 340 PRINT "QUANTOS HECTARES PRE
TENDE VENDER";
 341 INPUT Q: IF Q<0 THEN GO TO
850
 342 IF Q<A THEN GO TO 350
 343 GO SUB 720
 344 GO TO 340
 350 LET A=A-Q: LET S=S+Y*Q: LET
 C=0
 400 PRINT
 410 CLS : PRINT AT 18,0;"QUANTO
S HECTARES PRETENDE           DISPE
NSAR PARA ALIMENTAR           A POP
ULACAO ?";
 411 INPUT Q
 412 IF Q<0 THEN GO TO 850
 420 IF Q<=S THEN GO TO 430
 421 GO SUB 710
 422 GO TO 410
 430 LET S=S-Q: LET C=1: PRINT
 440 CLS : PRINT AT 18,0;"QUANTO
S HECTARES DESEJA SEMEAR ?";
 441 INPUT D: IF D=0 THEN GO TO
511
 442 IF D<0 THEN GO TO 850
 445 IF D<=A THEN GO TO 450
 446 GO SUB 720
 447 GO TO 440
 450 IF INT (D/2)<=S THEN GO TO
455
 452 GO SUB 710
 453 GO TO 440
 455 IF D<10*P THEN GO TO 510
 460 PRINT "MAS TEM APENAS";P;"P
ARA VIGIAR OS CAMPOS! ENTAO,"
 462 PAUSE 300
 470 GO TO 440
```

```
 510 LET S=S-INT (D/2)
 511 GO SUB 800
 515 LET Y=C: LET H=D*Y: LET E=0
 521 GO SUB 800
 522 IF INT (C/2)<>C/2 THEN GO T
O 530
 525 LET E=INT (S/C)
 530 LET S=S-E+H
 531 GO SUB 800
 533 LET I=INT (C*(20*A+S)/P/100
+1)
 540 LET C=INT (Q/20)
 542 LET Q=INT (10*(2*RND-.3))
 550 IF P<C THEN GO TO 210
 552 LET D=P-C: IF D>.45*P THEN
GO TO 560
 553 LET P1=((Z-1)*P1+D*100/P)/Z
 555 LET P=C: LET D1=D1+D: GO TO
 215
 560 CLS : PRINT "MATOU A FOME "
;D;" PESSOAS DURANTE UM ANO!!!"
 565 PRINT "PELO MAU GOVERNO"
 566 PRINT "E PELOS PREJUIZOS, V
AI SER             RETIRADO DO GOVER
NO"
 567 PRINT "E DECLARADO TRAIDOR
NACIONAL!!!": GO TO 990
 710 CLS : PRINT : PRINT "HAMURA
BI: PENSE DE NOVO!": PRINT "
 TEM APENAS"
 711 PRINT S;"  HECTOLITROS DE G
RAO, ENTAO,"
 712 RETURN
 720 CLS : PRINT "HAMURABI: PENS
E DE NOVO!": PRINT "   TEM APENA
S";A;"  HECTARES. ENTAO,"
 722 PAUSE 300
 730 RETURN
 800 LET C=INT (RND*5)+1
 801 RETURN
 850 PRINT : PRINT "HAMURABI: NA
O POSSO FAZER O QUE DESEJA."
 855 PRINT "PROCURE UM NOVO CONS
ELHEIRO!!!!"
 857 GO TO 990
 860 PRINT "AO FIM DESTES 10 ANO
S DE GOVERNO,";P1;" A PERCENTAGE
M DE"
 862 PRINT "POPULACAO MORTA PELA
 FOME POR ANO FOI DE"
 865 PRINT D1: LET L=A/P
 870 PRINT "COMECOU COM 10 HECTA
RES POR PESSOA E TERMINOU COM"
 875 PRINT L;"HECTARES.": PRINT
 880 IF P1>33 THEN GO TO 565
 885 IF L<7 THEN GO TO 565
 890 IF P1>10 THEN GO TO 940
 892 IF L<9 THEN GO TO 940
 895 IF P1>3 THEN GO TO 960
 896 IF L<10 THEN GO TO 960
 900 PRINT "FANTASTICO, ESPANTOS
O, NAO PODIA SER MELHOR!": GO TO
 990
 960 PRINT "PODERIA TER SIDO MEL
HOR, MAS"
 965 PRINT "REALMENTE NAO FOI MA
L DE TODO. ";INT (P*.8*RND);"PES
SOAS"
 970 PRINT "GOSTARIAMOS DE O VER
 ASSASSINADO MAS TEMOS OS NOSSOS
 PROBLEMAS  PARA RESOLVER."
 990 PRINT : FOR N=1 TO 10: PRIN
T CHR$ (7);: NEXT N
 995 PRINT AT 20,5;"ADEUS...por
agora !"
1000 PAUSE 400
1010 CLS
1020 PRINT AT 10,0;"para repetir
 o jogo....ENTER ou NEW LINE"
1030 IF INKEY$="" THEN GO TO 103
0
1040 RUN
```

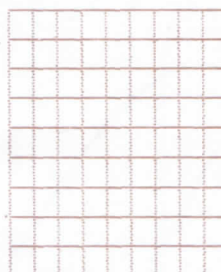
GRÁFICOS

ZX81

Adapt. de: FERNANDO JOSÉ/Caldas da Rainha
do LIVRO PILOTEZ VOTRE ZX81

```
 10 FOR f=0 TO 100
 20 PLOT 32+20*SIN (f/32*PI),22
+20*COS (f/32*PI)
 30 NEXT f
```

```
VALORES?
```

```
 10 LET L=40
 14 PRINT "HISTOGRAMA"
 16 PRINT
 20 PRINT "VALOR MAX."
 24 INPUT M
 26 CLS
 28 PRINT "VALORES?"
 30 INPUT A
 35 LET A=60*A/M
 37 LET L=L-2
 39 FOR N=1 TO A
 40 PLOT N,L
 42 NEXT N
 50 GO TO 30
```

```
 10 FOR N=0 TO 40 STEP 5
 20 FOR M=0 TO 40
 30 PLOT N,M
 40 PLOT M,N
 50 NEXT M
 60 NEXT N
```

BIORRITMO

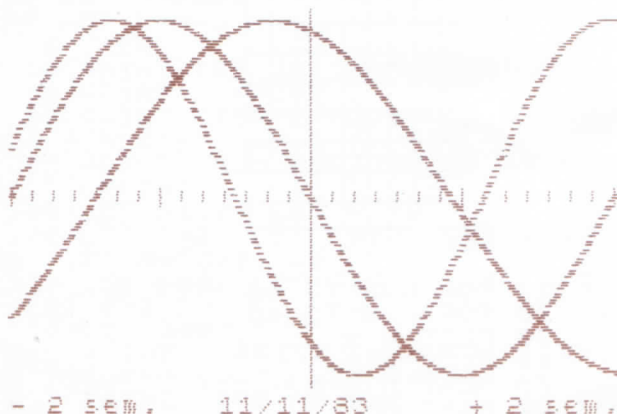
SPECTRUM 16 ou 48 K

Este programa demonstra o uso de subrotinas quando é necessário executar o mesmo tipo de operações em diferentes partes do programa.

Baseia-se na teoria de que o seu estado físico, emocional e intelectual segue ciclos regulares de 23, 28 e 33 dias respectivamente, a partir da data do seu nascimento.

Os gráficos destas 3 curvas são traçados para um período de 4 semanas.

Se o seu monitor for a cores, combine-as o melhor possível. Normalmente, tiram-se bons efeitos usando as cores magenta, verde, cyan e amarelo num fundo preto.

- 100 - 150 — Data de nascimento. Data actual. Cálculo do n.º de dias entre as duas datas.
- 160 - 190 — Cálculo aproximado do n.º de refeições e horas de sono.
- 200 — Pausa para leitura. Tecla «ENTER» para o traçado do gráfico.
- 210 - 280 — Apresentação do écran com cabeçalho e escala.
- 300 - 320 — Define as posições das 3 linhas em diferentes cores.
- 1000 — Subrotina para entrada da data, ano, mês e dia, usando o esquema «dd/mm/yyyy».
- 2000 — Desenho das curvas na escala.

```
 100 PAPER 7: INK 0: BORDER 7: INVERSE 0: OVER 0: FLASH 0: CLS
 110 PRINT AT 5,10; INK 3;"BIORRITMO": INPUT "O seu nome ?"; LINE n$
 120 PRINT AT 8,0;"Ola ";n$;AT 10,0;"Data de nascimento ?"
 130 GO SUB 1000: PRINT AT 10,19;d$: LET b$=d$: LET z=x
 140 PRINT AT 12,0;"Data do biorritmo ?": GO SUB 1000
 150 PRINT AT 12,17;"..";d$: LET d=x-z
 160 INK 3: PRINT AT 15,0;"Nesta data "
 170 PRINT "tem ";d;" dias,"
 180 PRINT "com ";3*d;" refeicoes,"
 190 PRINT "e com ";8*d;" horas de sono."
 200 INPUT """ENTER"" ver grafico"; LINE i$
 210 PAPER 0: INK 6: BORDER 0: CLS
 220 PRINT TAB 9;"BIORRITMO PARA"
 230 PRINT TAB 15-LEN n$/2;n$;TAB 12-LEN b$/2;"Nascido a ";b$
 240 FOR a=1 TO 255 STEP 9: PLOT a,73: DRAW 0,3: NEXT a
 250 FOR a=1 TO 255 STEP 63: PLOT a,71: DRAW 0,6: NEXT a
 260 PLOT 127,10: DRAW 0,128
 270 PRINT AT 21,0;"- 2 sem.";TAB 24;"+ 2 sem."
 280 PRINT AT 21,15-LEN d$/2;d$
 300 INK 3: PRINT AT 3,0;"Emocional": LET c=28: GO SUB 2000
 310 INK 4: PRINT AT 3,11;"Fisico": LET c=23: GO SUB 2000
 320 INK 5: PRINT AT 3,20;"Intelectual": LET c=33: GO SUB 2000
 330 PAPER 7: INK 0: BORDER 7
 340 STOP
1000 DATA 0,31,28,31,30,31,30,31,31,30,31,30,31
1010 INPUT "Ano ? ";y: LET x=365*y+INT (y/4)-INT (y/100)
1020 INPUT "Mes (1-12) ? ";m: IF m<1 OR m>12 THEN GO TO 1020
1030 RESTORE : FOR a=1 TO m: READ b: LET x=x+b: NEXT a
1040 LET l=y=4*INT (y/4) AND y<>100*INT (y/100)
1050 IF l AND m>2 THEN LET x=x+1
1060 READ b: IF l AND m=2 THEN LET b=29
1070 INPUT ("Dia(1-";b;") ?");d: IF d<1 OR d>b THEN GO TO 1070
1080 LET x=x+d: LET d$=STR$ d+"/"+STR$ m+"/"+STR$ y
1090 RETURN
2000 FOR a=0 TO 253: LET b=d-14+a/9
2010 PLOT a,74+60*SIN (2*PI*b/c): DRAW 2,0
2020 NEXT a
2030 RETURN
```

SPECTRUM LPRINT — Utilitário para impressora SINCLAIR/TIMEX

QUINTELA-SOFTWARE PESSOAL apresenta um utilitário que permite imprimir com mais de 32 colunas, numa impressora TIMEX ou SINCLAIR.

A título de demonstração, publicamos o calendário para 1984 e o descritivo deste utilitário.

Agradecemos que os leitores interessados nos comuniquem os seus comentários e níveis de interesse.

O programa "Spectrum LPRINT" é um utilitário que, uma vez carregado na memória do Spectrum (através da instução LOAD "" CODE) permite a escrita em linhas transversais nas impressoras Sinclair e TIMEX.

O programa cria um buffer interno onde a página em curso (cada página dispõe de 32 linhas com o número de colunas definido) é salvaguardada até que:

— se encontre cheia;
— seja mandada para a impressora (casos em que o texto entretando escrito será mandado para a impressora no formato definido);
— seja mandado limpar.

O acesso à impressão transversal faz-se através das instruções LPRINT e LLIST (respectivamente para a impressão de texto e para a listagem de programas) ou das instruções PRINT e LIST, se previamente se ligar o stream do ecrã (2) ao canal da impressora ('P'), designadamente através da instrução OPEN II 2, "p".

A maior limitação deste utilitário é o não reconhecimento das instruções de INVERSE e OVER. Igualmente não é reconhecido como tal o cursor left (CHR$ 8).

Quando inicialmente carregado o programa, a impressora é deixada no seu modo de funcionamento normal (o qual pode ser a todo momento "trocado" com a impressão lateral — N.B. a impressão lateral destrói o conteúdo do buffer normal da impressora, embora o contrário não se verifique). Para passar do modo de impressão normal para a impressão lateral use o comando LPRINT CHR$ 1. Para reverter à impressão normal use a instrução LPRINT CHR$ 0.

Pode definir o número de colunas que cada linha terá usando a instrução LPRINT CHR$ 3 + CHR$ X, onde X representa o número de colunas pretendido. O número mínimo de colunas é 16, o máximo é 128 e os números intermediários deverão ser sempre em passos de 16 (i. é, 32, 48, 64, 80, ...).

Para imprimir o texto existente no buffer interno (mesmo que ele não se encontre cheio — neste caso a impressão é automática), deve usar a instrução LPRINT CHR$ 12. Para limpar o conteúdo do buffer, sem imprimir, deve usar a instrução LPRINT CHR$ 11. Estas duas instruções definem a posição de escrita no início da página (linha primeira, coluna 0).

A instrução LPRINT CHR$ 10 permite o avanço de uma linha — line feed — sem modificação da coluna actual.

Fora as notas anteriores o utilizador não encontra diferenças sensíveis entre o normal funcionamento da impressora e o funcionamento em regime de impressão lateral. O comando AT, quando indique coluna superior ao numero máximo de colunas definido é, pura e simplesmente, ignorado. São igualmente ignorados os comandos de cor e os normalmente associados aos atributos (BRIGTH, INVERSE), bem como o comando OVER.

Recomenda-sé especial cuidado com a emissão pelo sistema de caracteres de controle (designadamente o CHR$ 13). Assim, a linha de programa: **10 LPRINT CHR$ 11** não coloca a posição de impressão na primeira, mas na segunda linha de escrita (o Spectrum encarrega-se, por si, de emitir, após o CHR$ 11, um CHR$ 13. A linha deveria ser reescrita como: **10 LPRINT CHR$ 11** Quando carregado o Spectrum fica colocado em modo de funcionamento normal; a posição PRINT no buffer interno está na primeira posição da página; o número de colunas definido é de 128.

ANO DE 1984 DC

Janeiro · Fevereiro · Março · Abril · Maio · Junho · Julho · Agosto · Setembro · Outubro · Novembro · Dezembro

USAR O SPECTRUM DE UM MODO DIFERENTE

COMO INSTRUMENTO DE MEDIDA: VU-METER

90% dos programas anunciados ou publicados para o SPECTRUM são, sem dúvida, jogos do tipo A ou B. Raramente aparece uma ou outra demonstração de criatividade no sentido de aplicação da máquina a outras áreas de actividade.

Não temos ilusões, por exemplo de que usar o SPECTRUM como VU-METER não seja apenas uma curiosidade para espantar os amigos; mas chama-se mais uma vez a atenção dos professores para a diversidade de situações em que o SPECTRUM pode ser actor principal.

O programa deve ser transcrito para a máquina com muito cuidado para não se cometer erros. Em seguida, execute o comando RUN e verá o ecran dividido em duas partes — na parte superior observará uma linha a ser desenhada.

Agora ligue uma fonte sonora. Por exemplo, faça passar uma cassete com som no seu gravador e accione a tecla PLAY. Previamente deve ter ligado o cabo do EAR/SPECTRUM ao EAR/Gravador. Ajuste o tom e o volume até que algum efeito se torne visível no ecran. Se nada acontecer, é porque algo está errado no programa ou na ligação.

Vamos agora explicar o que se passa.

A imagem na parte superior do ecran proporciona um registo dos últimos 255 valores que passaram pela posição de memória 23608. Logo que a figura ocupa todo o ecran, teremos a imagem «limpa» e renovamos o gráfico.

Sobre o «flash» que aparece na parte inferior da caixa:

— À esquerda teremos a simulação do VU-METER (existente em muitos equipamentos Hi-Fi). Os valores habituais vão de 0 da escala até que o «LED» fique vermelho.

— À direita, um pequeno bloco magenta (para os que trabalham a cores) deve iluminar-se quando os valores são muito altos, mostrando que a música ou o som estão particularmente altos.

Para executar o programa rapidamente, usamos uma pequena rotina em código máquina (25 byes) que lê o valor a partir do cabo do EAR e o transporta ao endereço 23608 (quem leu o capítulo 25 do «Spectrum Manual» sabe-lo-á). Normalmente é nessa posição que fica armazenado o aviso sonoro.

A própria rotina é automaticamente levada pelo programa para uma instrução REM no início do programa, o que implica a existência de mais que 30 caracteres de modo a haver espaço para o código. máquina.

Dado que esta rotina é recolocável, quem pretender evitar o efeito contrário na listagem do código máquina (use LIST 2 para evitar isso) deverá incluir o código máquina num outro local da memória. Preferimos guardar as rotinas em código máquina numa instrução REM porque se torna mais fácil localizá-las. Para quem tiver interesse, vamos apresentar as mnemónicas para o código máquina:

```
LD HL,(5C38)
LD A,00
LD(HL),A
LD B,FF
IN A,(FA)
CP FF
JR Z 01
INC(HL)
DJNZ-9
RET
NOP(x7)
```

Este círculo é executado 255 vezes e incrementa o valor da posição 23608 de cada vez que algo seja transmitido pelo EAR — obtem-se o último valor da posição 23608 sempre que a rotina seja chamada do BASIC pela linha 20. Aqui apresentamos um passo-a-passo de cada linha do programa:

1 REM zona do código máquina.
3 Verificação da transferência do código máquina.
5-8 Desenho do aparelho de medida.
9 Atribuir a a$, a string que é usada no simulador (20 graphic 5).
10 Ciclo inicial.
20 Chamada do código máquina.
21 Let a = valor correspondente à entrada no EAR e guardada na posição 23608.
30-40 Desenho da forma de onda.
51 Tradução do valor guardado na posição 23608.
60 Traçado da escala.
80 PRINT uma barra se o valor máximo for encontrado.
81 Pausa.
90 «Limpar» as barras.
200 Recomeço do ciclo.
8000— 9810Introdução do código máquina.

Quem pretender pode aperfeiçoar-se: que tal improvisar um «relógio» que mostre o valor, substituindo o «LED»? Pode usar o mesmo método seguido no manual para produzir o «relógio» ou talvez um mostrador digital.

Se for ainda mais ambicioso, tente converter todo o programa em código máquina o que melhorará a sua velocidade e rigor. De facto, tal como está, o programa fica a perder em «bits» de música de cada vez que actua o «LED» e desenha a metade superior do ecran.
A rotina em código máquina pode ser incorporada em vários programas que pretendam analisar o som. Poderá realizar-se um programa que apenas responda a certos sons provenientes do cabo do EAR (por razões de segurança) — DEIXAMOS AQUI A IDEIA.

OBSERVAÇÕES:

O PROGRAMA NÃO FUNCIONA COM A IMPRESSORA LIGADA. QUANDO FIZER RUN, A LINHA 1 FICA COM O ASPECTO APRESENTADO NA LISTAGEM.
LINHA 1 — 1 REM 1234567890123456789012345678901234567890.

In POPULAR COMPUTING
OUT./83

VU-METER

```
-20  -10  0  10  20          PEAK
▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒
```

```
   1 REM !8\>?@ COPY FLASH IF RE
TURN COPY (?4
   2>INK 0: PAPER 7: CLS
   3 IF PEEK 23760<>33 THEN GO S
UB 8000
   4 LET b$="
 "
   5 PRINT INK 7; PAPER 1;AT 0,0
;"    VU METER  "
   6 PLOT 0,30: DRAW 200,0: DRAW
 0,-30: DRAW 47,0: DRAW 0,30: DR
AW -47,0: PLOT 0,30: DRAW 0,-30:
 DRAW 200,0
   7 PRINT AT 19,1;"-20  -10   0
 10  20"
   8 PRINT AT 19,26; INK 0; PAPE
R 4;"PEAK"
   9 LET A$="▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒
▒"
  10 FOR N=0 TO 255
  20 LET (=USR 23760
  61 LET A=PEEK 23608
  29 INK 1
  30 PLOT N,87
  40 DRAW 0,(A/4)
  51 LET T=A/13
  52 INK 0
  54 IF t=0 THEN GO TO 70
  60 PRINT INK 5;AT 20,1;A$( TO
T)+B$( TO (20-T))
  61 PRINT AT 20,12; OVER 1; PAP
ER 7; INK 2
  70 PRINT AT 20,12; OVER 1; PAP
ER 7; INK 2
  80 IF T>14 THEN PRINT AT 20,26
; INK 3; PAPER 7;"▬▬▬▬"
  81 FOR F=0 TO 3: NEXT F
  90 PRINT AT 20,26;"     "
 200 NEXT N: RUN
8000 RESTORE
9600 FOR F=23760 TO 23784
9605 READ A: POKE F,A: NEXT F
9800 DATA 33,56,12,62,0,119,6,25
5,219,250,254,255,40,1,52,16,247
,201,0,0,0,0,0,0,0
9810 RETURN
```

CONVERSÃO

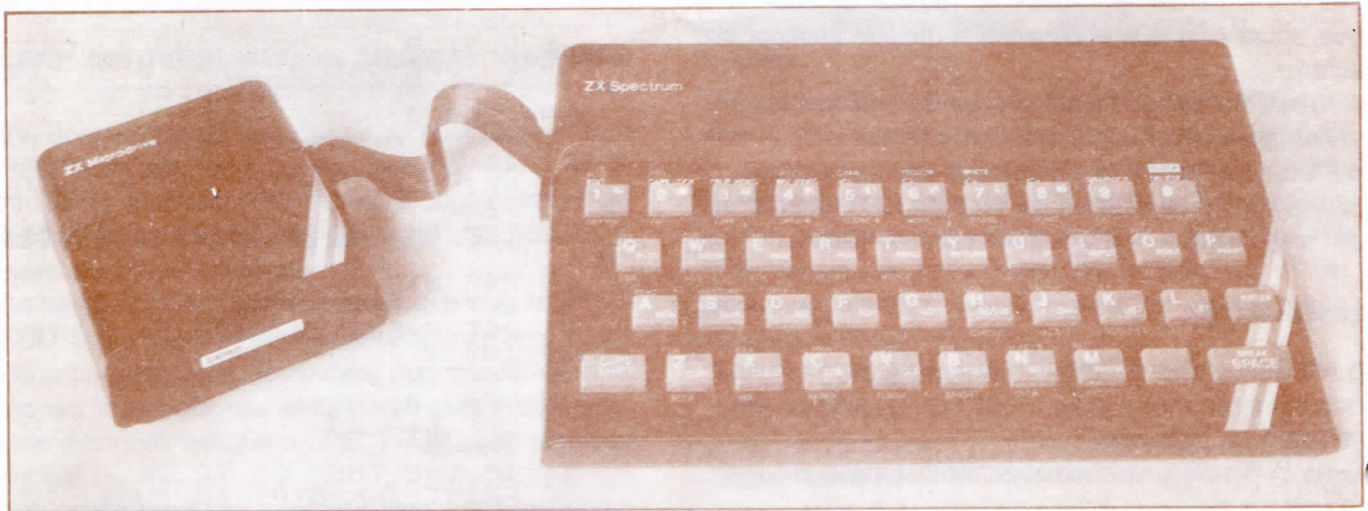
SPECTRUM

PROGRAMA ADAPTADO P/JOSE MARTINS

CONVERTE UM VALOR DE 8 BITS (EM BINARIO) NUM VALOR DECIMAL

```
   1 PRINT "PROGRAMA ADAPTADO P/JOSE MARTINS";AT 4,0;"CONVERTE UM VALOR DE 8 BIT
S (EM BINARIO) NUM VALOR DECIMAL": PAUSE 400: CLS
   5 REM "B/D"
  10 PRINT "VALOR BINARIO S0BRE 8 BITS?"
  20 INPUT B$
  30 CLS
  40 LET D=0
  50 FOR E=1 TO 8
  60 LET D=D+(VAL B$(E)*2^(8-E))
  70 NEXT E
  80 PRINT B$;" EM DECIMAL: ";D
  99 STOP
 100 REM COPYRIGHT 05/1983"
```

MICRODRIVE

SPECTRUM

Os novos dispositivos de extensão da capacidade do SPECTRUM, em termos de armazenamento de dados e programas, estão finalmente a ser entregues de acordo com as encomendas.

O atraso é grande e a SINCLAIR não parece disposta a fabricar muitos mais MICRODRIVES do que aqueles que foram encomendados pelos clientes ou distribuidores.

O Software para este periférico só deverá estar disponível durante o primeiro trimestre de 1984, dado que existe muita dificuldade em obter micro cassetes para reproduzir programas que sigam para o circuito comercial.

As últimas notícias referem que, em Londres, estarão disponíveis para venda a público em Fevereiro/Março. Em Portugal, os distribuidores exigem um sinal de 50% para registarem as encomendas de microdrives.

PREÇOS PREVISTOS

MICRODRIVE	11 000$00
INTERFACEE	6 000$00
CARTUCHOS (cada)	1 000$00

APLICAÇÃO

Basicamente, o microdrive destina-se a armazenar dados (gravar) ou a transferir dados (reproduzir) a partir de-ou para-uma memória externa: FLOPPY TAPE ou CARTUCHO.

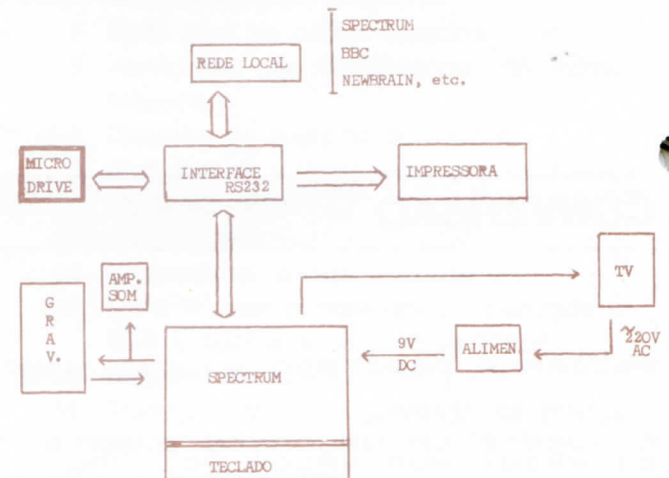
Quer isto dizer que o SPECTRUM ficará com a possibilidade de:

- Tratar dados na memória interna (16 ou 48 K RAM)
- Armazenar ou reproduzir dados em cassete.
- Armazenar ou reproduzir dados em cartucho (Floppy Tape).

Esta nova possibilidade traduz-se, na prática, por permitir a ampliação da memória (externa) para 85 Kbytes de dados (ou programas), que é a capacidade de cada FLOPPY TAPE. Teoricamente, existe mesmo a possibilidade de ligar 8 Microdrives simultaneamente (se bem duvidemos da eficácia deste tipo de ligação) e assim ter 680 Kbytes disponíveis.

A principal vantagem do uso do Microdrive será, eventualmente, o permitir acesso rápido a qualquer conjunto de dados armazenados no Floppy Tape, sem ter que se percorrer toda a fita de um extremo a outro, procurando o item desejado, como habitualmente sucede. (Cerca de 10 segundos o tempo médio de acesso.

SISTEMA

O SPECTRUM passará a ser o centro de um sistema expandido.

Os dados são provenientes de um dos dispositivos:

- TECLADO
- LEITOR DE CASSETES
- MICRODRIVE
- REDE LOCAL

e podem fluir para os seguintes dispositivos:

- GRAVADOR DE CASSETES
- MICRODRIVE
- IMPRESSORA
- REDE LOCAL
- AMPLIFICADOR DE SOM

(No próximo número continuaremos a falar do MICRODRIVE)

LIVROS — ACTUALIZAÇÃO DE PREÇOS

ZX81

TÍTULO	Preço (Fotocópias)
ZX81 BASIC BOOK	450$00
BYTEING DEEPER INTO YOUR ZX81	400$00
PEEK, POKE, BYTE AND RAM	300$00
THE ZX81 COMPANION	350$00
UNDERSTANDING YOUR ZX81 ROM	450$00
ZX81 A LA CONQUETE DES JEUX	325$00
PILOTEZ VOTRE ZX81	350$00
THE COMPLETE SINCLAIR ZX81 BASIC COURSE	650$00
LE PETIT LIVRE DU ZX81	350$00
NOT ONLY 30 PROGRAMMS FOR THE SINCLAIR ZX81	350$00
THE SINCLAIR ZX81 PROGRAMMING FOR REAL APPLICATIONS	425$00
MASTERING MACHINE CODE FOR YOUR ZX81	450$00
ZX FORTH USER'S MANUAL	200$00
MACHINE LANGUAGE MADE SIMPLE FOR YOUR SINCLAIR	375$00
SINCLAIR ZX81 ROM DISASSEMBLY (PART A+B)	400$00
THE ZX81 POCKET BOOK	350$00
LA CONDUITE DU ZX81	300$00
THE EXPLORERS GUIDE TO THE ZX81	450$00
THE GATEWAY GUIDE TO THE ZX81 and ZX80	460$00
GETTING ACQUAINTED WITH YOUR ZX81	350$00
WHAT CAN I DO WITH 1 K — 40 PROGRAMS & ROUTINES	325$00
HINTS & TIPS FOR THE ZX81	250$00

SPECTRUM

TÍTULO	Preço (Fotocópias)
SPECTRUM HARDWARE MANUAL	300$00
THE SPECTRUM PROGRAMMER	375$00
GAMES TO PLAY ON YOUR ZX SPECTRUM	230$00
OVER THE SPECTRUM	410$00
BETTER PROGRAMMING FOR YOUR SPECTRUM AND ZX81	710$00
THE ZX SPECTRUM AND HOW TO GET THE MOST FROM IT	325$00
60 GAMES AND APPLICATIONS FOR THE ZX SPECTRUM	250$00
THE HOBBIT (ROMANCE EM QUE SE BASEOU O PROG.)	720$00
PROGRAMMING YOUR ZX SPECTRUM	600$00
20 BEST PROGRAMMS FOR THE ZX SPECTRUM	325$00
EASY PROGRAMMING FOR THE ZX SPECTRUM	350$00
INTRODUCING SPECTRUM MACHINE CODE	400$00
LA PRATIQUE DU ZX SPECTRUM (VOL 1)	420$00
LA PRATIQUE DU ZX SPECTRUM (VOL 2)	420$00
THE SPECTRUM HANDBOOK	540$00
40 BEST MACHINE CODE ROUTINES FOR THE ZX SPECTRUM	360$00
SPECTRUM GRAPHICS	500$00
THE SPECTRUM BOOK OF GAMES	375$00
UNDERSTANDING YOUR SPECTRUM	500$00
THE WORKING SPECTRUM	550$00
THE ZX SPECTRUM EXPLORED	550$00
THE COMPLETE SPECTRUM ROM DISASSEMBLY	600$00
GAMES FOR YOUR ZX SPECTRUM	325$00

DESCONTO DE 10% PARA SÓCIOS DO CLUBE

SOFTWARE ZX81

— POWER PACK 1 (Educacional) .... 450$00
— SIMULADOR DE VOO (Jogo) .... 550$00
— FORTALEZA DE ZORLAC (Jogo) .... 450$00
— BACKGAMMON (Jogo) .... 450$00
— THE GAMBLER (Jogo) .... 450$00
— SUPERMAZE (Jogo) .... 450$00
— GRIMM'S FAIRY TRAILS (Jogo) .... 450$00
— XADREZ AUS (Jogo) .... 600$00
— SUPER MATEMÁTICA (Educ.) .... 320$00
— VU-CALC (Gestão) .... 550$00

☆ DESCONTO DE 15% PARA SÓCIOS DO CLUBE ☆

SOFTWARE SPECTRUM

- **EDUCAÇÃO**

— GEOMETRIA DESCRITIVA
(Nomenclatura e Testes. Com gráficos) .... 600$00

— EQUAÇÃO DA RECTA
(Com gráficos) .... 600$00

— MATEMÁTICA II
(Números primos; M.D.C.; M.M.C.; operações c/ fracções) .... 600$00

- **UTILITÁRIOS**

— ZX SLOWLOADER (Permite passar cassetes gravadas c/ prog. do ZX81 para o Spectrum. No final da Leitura, o programa que fica no Spectrum pode ser gravado e corrigido) ..... 800$00

— SPECTRUM LPRINT (Permite escrever na impressora com um número de colunas superior a 32 e até ao máximo de 128, e no sentido vertical) .... 400$00

— GAMES DESIGN (Gerador de figuras e animação para os jogos que o utilizador pretende criar em termos de desenho e movimento. Possui 8 jogos já prontos como exemplo. .... 1 000$00

- **JOGOS**

— TURBO DRIVER (Corrida de automóveis) .... 400$00
— HARRIER ATTACK (Batalha de aviões) .... 400$00
— AQUAPLANE (Ski aquático) .... 400$00
— AUTOMONOPÓLIO (Monopólio) .... 400$00
— MANIC MINER (Jogo animado no interior de 1 túnel) .... 400$00

- **LINGUAGENS**

— FORTH (C/ vírgula flutuante. Compilador) .... 600$00

☆ DESCONTO DE 20% PARA SÓCIOS DO CLUBE ☆

PERGUNTAS / RESPOSTAS / SUGESTÕES

— «Como posso substituir a ordem SCROLL do ZX81 para adaptar um programa para o ZX SPECTRUM?»

LUÍS FILIPE
Caldas da Rainha

Quando encontrar uma instrução SCROLL para ZX81 e pretender adaptar o mesmo programa para o SPECTRUM, páre e simplesmente omita esta instrução.

— HUGO ASSUMPÇÃO/Lisboa enviou-nos um programa que "sintetiza e melhora os programas das págs. 29 e 30 da revista n.° 12" — CÁLCULO DE ÁREAS.

```
   5 GO TO 50
  10 LET Y=VAL X$
  15 RETURN
  50 PRINT TAB 5;"CALCULO DE ARE
AS"
  70 GO SUB 500: GO SUB 550: GO
SUB 700
  80 PRINT AT 9,0,,
  85 LET P=0: LET H=C
 100 LET R=(B-A)/H: LET X=A
 130 GO SUB 10
 140 LET Q=0: LET Q=Q+Y: LET X=X
+R
 160 GO SUB 10
 170 LET Q=Q+Y: LET H=H-1: LET P
=P+Q*R/2
 200 IF H<>0 THEN GO TO 140
 220 PRINT AT 5,0;"METODO DOS ";
"METODO DE ";AT 7,0;"TRAPEZIOS",
"SIMPSON"
 225 PRINT AT 9,0;P
 230 LET P=0: LET H=C: LET R=(B-
A)/2/H: LET X=A
 250 GO SUB 10
 260 LET Q=0: LET Q=Q+Y: LET X=X
+R
 270 GO SUB 10
 280 LET Q=Q+Y*4: LET X=X+R
 290 GO SUB 10
 300 LET Q=Q+Y: LET H=H-1: LET P
=P+Q*R/3
 320 IF H<>0 THEN GO TO 260
 330 PRINT AT 9,16;P
 340 INPUT "MAIS INTERVALOS ?";M
$: IF M$="S" THEN GO SUB 700: GO
 TO 80
 355 INPUT "PRETENDE ALTERAR A F
UNCAO ?";M$: IF M$="S" THEN GO S
UB 500: GO TO 80
 360 INPUT "PRETENDE ALTERAR O I
NTERVALO ?";M$: IF M$="S" THEN G
O SUB 550: GO TO 80
 365 STOP
 500 INPUT "Funcao de X ?";X$
 505 PRINT AT 1,5;"Funcao:";X$
 510 RETURN
 550 INPUT "VALOR DA ABCISSA INI
CIAL ?";A
 555 INPUT "VALOR DA ABCISSA FIN
AL ?";B
 560 PRINT AT 2,5;"ENTRE:";A;" E
 ";B
 565 RETURN
 700 INPUT "NUMERO DE INTERVALOS
 ?";H
 705 PRINT AT 3,5;"COM ";H;" INT
ERVALOS "
 706 LET C=H
 710 RETURN
```

NOVOS PROGRAMAS PARA O SPECTRUM

POR ENGANO DA N/ PARTE, OS PREÇOS DE ALGUNS PROG. FORAM ERRADAMENTE INDICADOS.

ASSIM:

— JOGOS, AVENTURAS E ASTROLOGIA .... 400$00 (e não 300$00)

— DEVPAC (utilitário) .... 1000$00 (e não 600$00)

MONITOR PARA CÓDIGO MÁQUINA

PÁG. 22 LINHA 1030:

IFi>65535 THEN LETi=S+10 * I+9: GOTO 150

POR FALTA DE ESPAÇO NESTE NÚMERO, NÃO INCLUÍMOS A HABITUAL RUBRICA **ESPAÇO SPECTRUM.**

PEDIMOS A COMPREENSÃO DE TODOS, NOMEADAMENTE DO SEU AUTOR *"FERNANDO PRECES"* QUE TÃO ACTIVAMENTE TEM COLABORADO COM O **"CLUBE Z 80".**

O PROG. «FIRE FOX» PUBLICADO NO N.° ANTERIOR TEM ALGUMAS GRALHAS. PUBLICAREMOS NO PRÓXIMO N.° A LISTAGEM CORRECTA.